A PROBABILISTIC DIGITAL SIGNATURE METHOD

[0001] This disclosure is based upon, and claims priority from, French Application No. 00/03918, filed March 28, 2000, the contents of which are incorporated herein by reference.

Background of the Invention

[0002] The present invention concerns a method of generating probabilistic digital signals to enable the integrity of a transmitted message to be checked.

[0003] The present invention applies in particular to the field of chip cards, e.g. smart cards, with or without contacts. Such cards constitute protected information carriers and generally include a microcontroller incorporated on an integrated-circuit chip. A microcontroller has an architecture similar to that of a computer. It has a processing unit consisting of a microprocessor or CPU associated with different types of memory. A non-volatile memory, of the ROM type for example, generally has at least one program for implementing a signature algorithm.

[0004] The invention applies in particular to algorithms for generating and checking digital signatures. The objective of such algorithms is to calculate one or more integers, in general a pair, referred to as the signature and associated with a given message in order to certify the identity of the signature and the integrity of the signed message. Such algorithms make it possible on the one hand to generate signatures and on the other hand to check these signatures.

[0005] The signature is said to be probabilistic when the algorithm uses a random number in the generation of the signature, this random number being secret and regenerated with each new signature. Thus one

and the same message transmitted by the same user can have several distinct signatures.

[0006] An example of such an algorithm can be illustrated by the DSA (Digital Signature Algorithm).

The parameters of the DSA are:

- p, a large known prime number, of 512 or 1024 bits,
- q, a prime number which divides p-1, of 160 bits,
- g, an integer chosen such that $g_q = 1 \bmod p$

  with $g \neq 1 \bmod p$.

[0007] The secret key x is a randomly fixed number between 0 and 2^{160}-1, and the public key is related therein to x by the equation $y = g^x \bmod p$.

[0008] The message to be sent is identified as m. The DSA signature of m is the pair (r,s) defined as follows:

$r = (g^k \bmod p) \bmod q$;

$s = (h(m) + r\,x) / k \bmod q$;

with k a random number of 160 bits such that $k<q$, regenerated with each signature,

and h(m) the initial message m encoded by means of a chopping function which is a pseudo-random cryptographic function.

[0009] The signature is verified as follows:

A first intermediate calculation is performed

$w = s^{-1} \bmod q$.

It is checked whether $((g^{w \cdot h(m)}\ y^{r \cdot w}) \bmod p) \bmod q = r$.

If this equality is true, the signature is authentic.

[0010] The signature (r,s) was generated with the secret key x and a secret random number k different for each signature, and it was checked with the public key y. Thus anyone can authenticate a card and its bearer without holding its secret key.

[0011] The use of the chopping function in generating the signature is found in almost all probabilistic signature generating algorithms based on a discrete logarithm calculation. It makes it possible to guarantee the non-reproducibility of the signature by breaking its linearity.

[0012] The use of this chopping function nevertheless has drawbacks since it assumes firstly that this function h behaves like a random function, which is not always true, and secondly that this function h is implemented in the memory of the integrated-circuit chip of the protected device (the chip card for example). However, the code size necessary for implementing the chopping function is very high, approximately 1 to 2 kilobytes.

[0013] The economic constraints related to the chip card market require constant research with a view to controlling its cost. This effort often consists of the use of simpler components. In such a context, the implementation of public key algorithms on inexpensive microcontrollers of the 8-bit type with an 8051 (Intel) or 6805 (Motorola) kernel, for example, represents an increasing advantage. It is, however, not possible to implement a digital signature algorithm such as the DSA or of the same type having recourse to a chopping function on such microcontrollers.

[0014] The aim of the invention is to resolve these constraints by proposing a solution which is adapted to microcontrollers having few calculation resources.

Description of the Invention

[0015] The object of the present invention is a method of generating probabilistic digital signatures which makes it possible to dispense with the chopping function, without impairing the security of the messages exchanged.

[0016] To this end the invention proposes a method for transforming a probabilistic signature algorithm using a chopping function into another algorithm not using this function. To this end, the initial probabilistic algorithm is used twice instead of once to sign the message directly, that is to say the initial unchopped message. In this way twin signatures associated with the same message are generated.

[0017] The invention concerns more particularly a method relating to probabilistic digital signatures of a message, between a signatory and a checker, using an algorithm based on the calculation of a discrete logarithm. The method includes the step, for the signatory, of generating at least two signatures for the same unchopped message, these signatures being calculated by the algorithm by means of the same public and private key parameters using respectively distinct random values. The method further includes, for the checker, the step of checking all the signatures of the message.

[0018] According to one application, the probabilistic algorithm is the DSA (Digital Signature Algorithm).

[0019] According to another application, the probabilistic algorithm is the Schnorr algorithm.

[0020] The invention advantageously applies to any protected device of the chip card type, and in particular to devices having an 8-bit microcontroller.

[0021] The method according to the invention has the advantage of dispensing with the chopping function and thus minimizing the memory utilization. In addition, the calculation speed is increased, even if a double calculation is required. This is because using a chopping function is tricky on simple 8-bit microcontrollers, which are inexpensive and are often being used more and more in order to contain the manufacturing costs of the devices.

[0022] In addition, the method according to the invention guarantees security in the execution of any probabilistic digital signature generating algorithm.

[0023] The description refers to the DSA signature algorithm, but also applies to all other probabilistic signature algorithms and to their variants such as El Gamal, Schnorr, EC-DSA or Abe-Okamoto, for example, which also use the chopping function in generating pairs of signatures.

[0024] The signature generation method according to the invention is based on the calculation of at least two signatures, which are then referred to as twins, for the same initial unchopped message m. The signature thus comprises at least two signatures calculated by means of the same public key y and private key x parameters using respectively distinct random numbers k_1, k_2, ... k_n.

[0025] The message signature thus becomes $(r_1,s_1,r_2,s_2,\ldots r_n,s_n)$, with the n pairs (r_1,s_1) (for i ranging from 1 to n) calculated and checked in accordance with the conventional signature generation and checking methods, whether it is a case of the DSA, Schnorr or any other algorithm using a chopping function.